

Prerequisites

The following assumes you are able to build OpenCPN from source and have the needed tools. You can find the detailed instructions at http://opencpn.org/compiling_source_linux and http://opencpn.org/compiling_source_windows

Git magic

If you followed the building instructions above, you already have the local repository clone and all the needed tools on your computer. Open a terminal (run cmd.exe) and change to the directory where you have cloned the source.

Personal advice: to save a lot of problems later, don't be afraid to use a lot of small per-task branches. Could sound weird especially to the guys used to the traditional version control systems, but this is how it's meant to be with git and modern distributed version control systems. And believe me that merging branches is much easier than cherry-picking commits belonging to a single task.

Following are a few git commands you will for sure need sooner or later:

To update to the latest upstream code

```
git pull origin
```

To see all the branches in your local repository

```
git branch
```

To create a new branch

```
git branch [NAME]
```

To change to another branch (don't forget to commit or reset your changes before changing to another branch, otherwise git will not be happy throwing away your work and overwriting it with other branch)

```
git checkout [NAME]
```

To delete a branch

```
git branch -D [NAME]
```

To add a file to the current branch

```
git add [filename.ext]
```

To delete a file from the current branch

```
git rm [filename.ext]
```

To commit your local changes to the current branch

```
git commit -a -m "My commit message text"
```

To throw away the latest code changes you've done and get back to the last committed state

```
git reset --hard
```

Be aware that this is not a Git tutorial at all. To know how to use Git, consult the Git book at <http://book.git-scm.com/> or any other tutorials on the net.

Code formatting

Please read the page on [Code Formatting](#) before writing new code for OpenCPN.

Publishing your code

The best way to ensure your code gets properly submitted to OpenCPN is to create a clone of the OpenCPN GitHub master repository, and push your commits to your cloned repository. When everything is available on Github you can either send a mail the developers or create an issue in <http://opencpn.org/flyspray/> describing your work and referenceing the github commit URL.

The official development master is here: <https://github.com/OpenCPN/OpenCPN>

Creating a patch

Expecting your branch is checked out and you are in your working directory. Simply issue:

```
git diff master> mystuff.patch
```

In the newly created file called *mystuff.patch* you may review the code you are going to submit with any text editor.

Submitting a patch

Create a Flyspray account at <http://opencpn.org/flyspray/>

Bugfix:

Attach the patch with comment to the corresponding issue in the bugtracker - for 2.2.x
<http://opencpn.org/flyspray/index.php?project=3>

New feature:

Attach the patch with comment to the corresponding issue in the Feature requests tracker at
<http://opencpn.org/flyspray/index.php?project=6>

From:
<https://opencpn.org/wiki/dokuwiki/> - **OpenCPN Manuals**

Permanent link:
https://opencpn.org/wiki/dokuwiki/doku.php?id=opencpn:developer_manual:developer_guide:prerequisites:prerequisites

Last update: **2017/01/06 19:18**

